

## **REMARKS**

In response to the Final Office Action mailed July 28, 2009, the Assignee respectfully requests reconsideration.

### **I. Summary of Telephone Conference with Examiner**

The Assignee's representatives Daniel T. Wehner and Richard F. Giunta thank Examiner Lerner for the courtesies extended in granting and conducting a telephone interview on October 15, 2009, the substance of which was summarized herein. As discussed in more detail below, a distinction between configuring error recovery options for a dialogue module of Marx and applying a selected catch style to each of the plurality of catch events in an application as recited in each of the independent claims was discussed. The Examiner appreciated this distinction and indicated that amendments made to clarify the distinction would appear to distinguish over the cited references.

### **II. Rejections Under 35 U.S.C. §103**

The Office Action rejects claims 1-10 and 30-39 (including independent claims 1, 30 and 39) under 35 U.S.C. §103(a) as allegedly being obvious over U.S. Patent No. 6,173,266 ("Marx") in view of U.S. Patent No. 7,266,181 ("Zirngibl").

#### **A. Overview of Some Embodiments**

Programmers of interactive speech applications often are faced with managing and preparing audio responses to catch events that occur during use of the speech application (§0002). Representative catch events include user requests for help, non-input entries (in which no user response is received), and non-matching entries (in which the user response is not understood) (§0002).

In some embodiments, a style-selection menu that enables a programmer to select a particular catch style for an interactive speech application may be presented to the programmer (§0020; Fig. 1). When creating the interactive speech application, the selected catch style may be applied to each of the catch events in the interactive speech application (§0019). Selection of a single catch style for all catch events in the speech application obviates the need for creation of text

and/or code for each individual catch event, thereby simplifying the programmer's task in managing catch events (¶¶0018-0019; Fig. 1].

The foregoing summary is provided to assist the Examiner in appreciating some aspects and/or applications of embodiments described in the present application. However, this summary may not apply to each of the independent claims, and the language of the independent claims may differ in material respects from the summary provided above. Thus, the Assignee respectfully requests that careful consideration be given to the language of each of the independent claims and that each be addressed on its own merits, without relying on the summary provided above. In this respect, the Assignee does not rely on the summary provided above to distinguish any of the claims over the prior art. Rather, the Assignee relies only upon the arguments provided below.

B. Discussion of Marx

As discussed during the telephone conference, Marx is directed to a graphical method for developing an interactive speech application (abstract). To create a call flow, an application developer links together prepackaged software modules called "dialogue modules" that each represents a discrete dialogue task in an interactive speech application (Col. 4, lines 20-32; Fig. 7). Each dialogue module has customizable parameters, which allow the application developer to enable/disable certain features such as "barge-in," or to select appropriate error handling methods and prompts (Col. 4, lines 33-49; Fig. 9). The complexity of the processing in each dialogue module varies depending on the discrete dialogue task assigned to the dialogue module (Col. 10, lines 42-44).

As described above, dialogue modules include error recovery methods which can be customized by an application developer for specific instances of the dialogue module (Col. 13, lines 12-14). For example, the error recovery window 1600 shown in Fig. 16 allows an application developer to "customize the error recovery properties to determine the call flow within a Dialogue Module instance" (Col. 20, lines 17-21). That is, *for each instance of a dialogue module in the interactive speech application*, an application programmer specifies how the system will respond to errors during the portion of the call flow in which the dialogue module is executing.

C. The Combination of Marx and Zirngibl Fails to Disclose All Limitations of Independent Claims 1, 30, and 39

---

*i. Independent Claim 1*

Amended claim 1 recites, *inter alia*, “upon selection of a catch style, preparing the system’s audio response for **the plurality of catch events in the interactive voice application**” (emphasis added). Support for this amendment is found at ¶¶0029-0030 of the specification as originally filed. As discussed during the telephone conference, neither Marx nor Zirngibl discloses or suggests this limitation of claim 1.

The Office Action appears to assert that the default error recovery methods for the dialogue module templates of Marx read on “catch styles” as recited in claim 1 (Office Action, pages 3-4). However, as described above, an application developer in Marx must define error recovery methods for individual dialog modules rather than selecting a “catch style” which is applied to multiple catch events in an interactive voice application. Because a programmer in the system of Marx is unable to select a single catch style that is applied to multiple catch events in an interactive voice application, but instead must separately configure each dialogue module in the application, it is improper to consider the error recovery methods defined in each of the dialogue modules to be catch styles as recited in claim 1.

Furthermore, even if it were proper to consider the default error recovery parameters of Marx to be a catch style, which the Assignee does not concede, claim 1 requires presenting a style-selection menu *for a plurality of catch styles* that allows for selection of one or more of the catch styles. Marx does not present a plurality of default error recovery parameter styles in a style-selection menu. Rather, the application developer in Marx decides *for each dialogue module* whether to use the default error recovery parameters defined in the baseline configuration or to customize the error recovery parameters for processing during the task defined by the dialogue module (Marx, col. 20, lines 17-32; Fig. 16).

The Office Action asserts that the GUI shown in Fig. 7 of Marx is such a style-selection menu, and that Marx discloses that at least some of the dialogue modules that can be selected using the GUI are directed to dealing with error conditions (Office Action, pages 9-10). However, the Assignee respectfully disagrees with this interpretation of Marx. As discussed above, each of the

dialogue modules in Marx represents a specific task in a call flow for a speech application and the error recovery parameters during performance of the specific task are configurable for particular instances of a dialogue module (Marx, col. 13, lines 12-14). That is, because the error recovery parameters for each dialogue module must be individually configured, it is improper to consider the different dialogue modules of Marx to be different “catch styles” which may be selected and applied to a **plurality of catch events in an interactive voice application**, as required by claim 1.

Zirngibl fails to cure these deficiencies of Marx as Zirngibl also fails to disclose or suggest “applying [a] selected catch style to a plurality of catch events” in an interactive voice application, as recited in claim 1. The Office Action appears to assert that the XSL style sheets of Zirngibl read on catch styles as recited in claim 1 (Office Action, page 9). However, as described in a prior response dated June 24, 2009, it is improper to consider the XSL style sheets of Zirngibl to be catch styles as recited in claim 1 because although XSL style sheets relate to a style that dictates the layout of a document, the XSL style sheets are not selectable from a style-selection menu nor are they applied to a plurality of catch events in an interactive voice application as required by claim 1.

For at least the foregoing reasons, claim 1 patentably distinguishes over the combination of Marx and Zirngibl and it is respectfully requested that the rejection under 35 U.S.C. §103 be withdrawn. Claims 2-10 depend from claim 1 and each is allowable for at least the same reasons.

#### *Independent Claim 30*

For reasons that may be appreciated from the foregoing discussion, the combination of Marx and Zirngibl does not disclose or suggest, “...each catch style defining a system response **the plurality of catch events in the speech application...** (emphasis added)” as recited in amended claim 30. For at least this reason it is respectfully requested that the rejection under 35 U.S.C. §103 be withdrawn.

#### *Independent Claim 39*

For reasons that may be appreciated from the foregoing discussion, the combination of Marx and Zirngibl does not disclose or suggest, “... preparing, upon selection of a catch style, the system’s audio response for **the plurality of catch events in the speech application by applying**

**the selected catch style to the plurality of catch events ... (emphasis added)**” as recited in amended claim 39. For at least this reason it is respectfully requested that the rejection under 35 U.S.C. §103 be withdrawn.

III. New Dependent Claims 40 and 41

Claims 40 and 41 are newly added to this application. Support for these new dependent claims is found at ¶0029 of the specification as originally filed. Each of these claims is believed to be in allowable condition at least because it depends from a claim that patentably distinguishes over the cited art of record.

IV. General Comments on Dependent Claims

Since each of the dependent claims depends from a base claim that is believed to be in condition for allowance, for the sake of brevity, the Assignee believes that it is unnecessary at this time to argue the further distinguishing features of the dependent claims. However, the Assignee does not necessarily concur with the interpretation of the previously presented dependent claims as set forth in the Office Action, nor does the Assignee concur that the basis for rejection of any of the previously presented dependent claims is proper. Therefore, the Assignee reserves the right to specifically address the further patentability of the dependent claims in the future.

**CONCLUSION**


In view of the foregoing amendments and remarks, this application should now be in condition for allowance. A notice to this effect is respectfully requested. If the Examiner believes, after this amendment, that the application is not in condition for allowance, the Examiner is requested to call the Assignee's representative at the telephone number indicated below to discuss any outstanding issues relating to the allowability of the application.

The Assignee believes no fee is due with this response. However, if a fee is due, please charge our Deposit Account No. 23/2825 under Docket No. N0484.70570US00 from which the undersigned is authorized to draw.

Dated: October 28, 2009

Respectfully submitted,

By:

  
Daniel T. Wehner, Ph.D., Reg. No: 63,480  
Richard F. Giunta, Reg. No: 36,149  
Wolf, Greenfield & Sacks, P.C.  
Federal Reserve Plaza  
600 Atlantic Avenue  
Boston, Massachusetts 02210-2206  
Telephone: (617) 646-8000